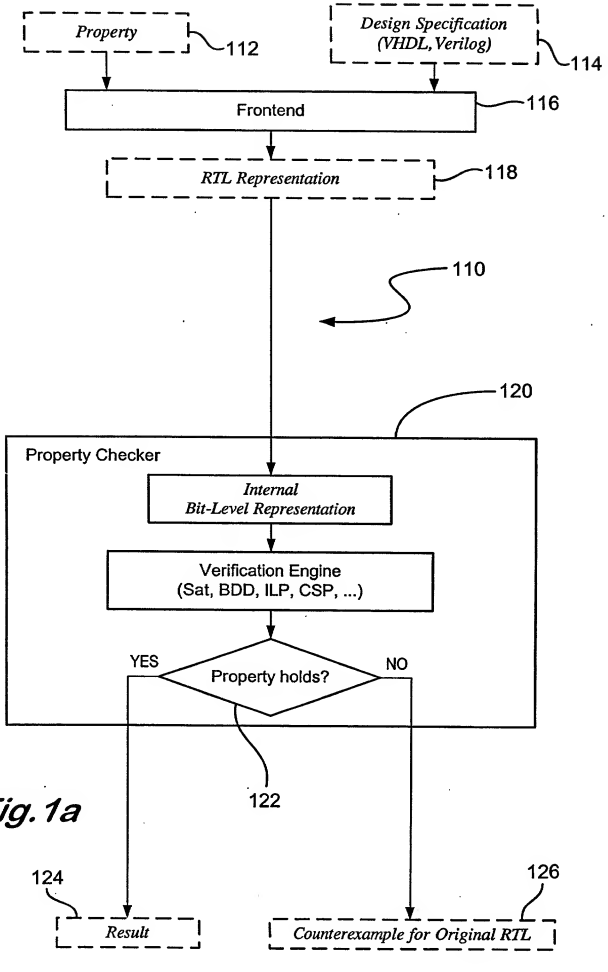


Fig. 1a

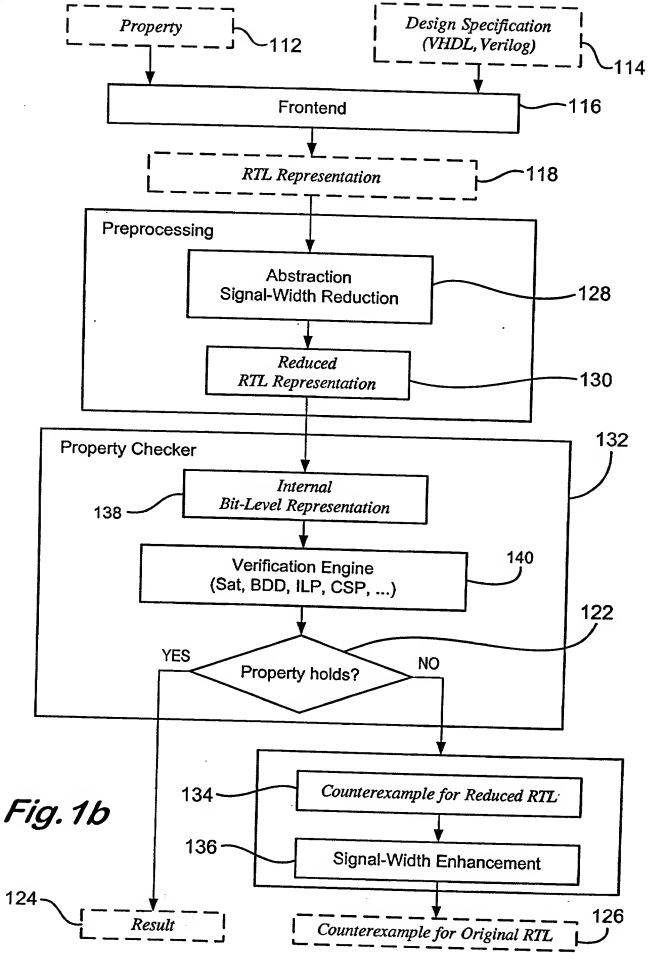
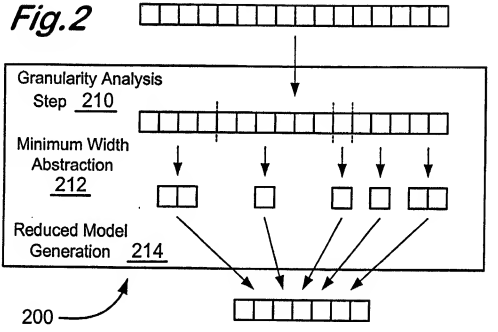


Fig. 1b

Fig.2



Original Signal $x_{[16]}$
216

Chunks : $x_{[16]}[15,12], x_{[16]}[11,6], x_{[16]}[5,5], x_{[16]}[4,2], x_{[16]}[1,0]$
218

Minimum Widths: 2 + 1 + 1 + 1 + 2 = 7
220

Corresponding Reduced Signal $x'_{[7]}$
222

Fig.4

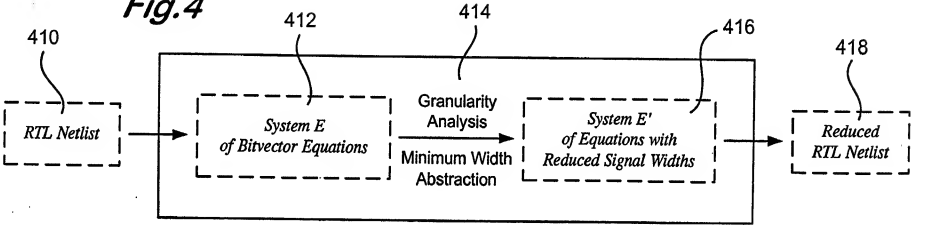


Fig.3

Bitvector Operator	Syntax	Example
bitvector variables bitvector constants	$x_{[n]}$ $c_{[m]}$	$x_{[8]}, y_{[1]}, z_{[4]}, \ldots$ $10011_{[5]}, 00111111_{[8]}, 0_{[1]}, 1_{[1]}, \ldots$
concatenation	$\otimes$	$x_{[16]} \otimes y_{[4]}$
extraction	$[j, i]$	$x_{[8]}[5, 2]$
bitwise negation (inversion)		neg $(x_{[8]})$
bitwise Boolean operations	and, or, xor nand, nor, xnor	$x_{[12]}$ and $y_{[12]}$, $x_{[12]}$ or $y_{[12]}$, $x_{[12]}$ xor $y_{[12]}$ $x_{[12]}$ nand $y_{[12]}$, $x_{[12]}$ nor $y_{[12]}$, $x_{[12]}$ xnor $y_{[12]}$
if-then-else	ite	ite$(a_{[4]} = b_{[4]}, x_{[8]}, y_{[8]})$ ite$(a_{[4]} < b_{[4]}, x_{[8]}, y_{[8]})$
arithmetic	$+, -$ $*$	$x_{[32]} + y_{[32]}, x_{[32]} - y_{[32]}$ $x_{[16]} * y_{[16]}$
memory read memory write	$mem_{[m \cdot n]}[i_{[l]}]$	$x_{[10]} := mem_{[128 \cdot 10]}[i_{[7]}]$ $mem_{[32 \cdot 8]}[i_{[5]}] := x_{[8]}$

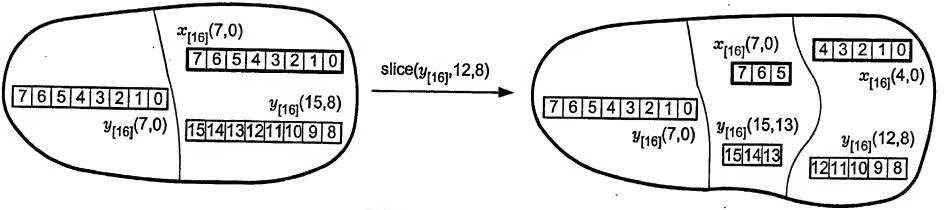


Fig.5

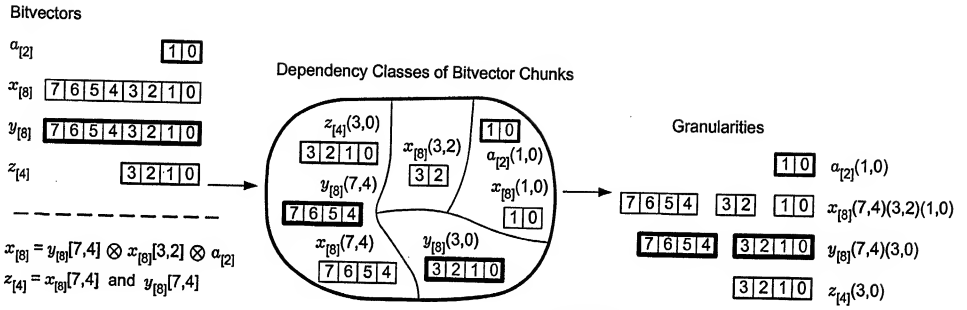


Fig.6

Fig.6a

Process 1 Granularity Analysis of Bitvector Equations

```
1  gran(e) {
2    switch (e);
3    case e ≡ 's[n] = t1[m1] ⊗ t2[m2]':
4    case e ≡ 's[n] = (t1[m1] ⊗ t2[m2])[j, i]':
5                                          gran('s[n][n−1, m2] = t1[m1]'); gran('s[n] [m2 −1,0] = t2[m2]');
6                                          if (j<m2) {
7                                            gran('s[n] = t2[m2][j, i]');
8                                          } else if (i ≥ m2) {
9                                            gran('s[n] = t1[m1][j−m2, i−m2]');
10                                         } else {
                                             gran('s[n] = t1[m1][j−m2, 0] ⊗ t2[m2][m2−1, i]');
                                           }
11   case e ≡ 's[n] = (t[m][l,k])[j,i]':   gran('s[n] = t[m][k + j, k + i]');
12   case e ≡ 's[n] = ite(a[m] = b[m], t[t[n]], te[n])[j,i]':
                                           gran('s[n] = ite(a[m] = b[m], t[t[n]][j,i], t[e[n]][j,i])');
13   case e ≡ 's[n] = (t1[m] and t2[m])[j, i]':
                                           gran('s[n] = t1[m][j,i] and t2[m][j, i]');
14   ...
15   case e ≡ 's[n] = neg(t[n])':          gran('s[n]=t[n]');
16   case e ≡ 's[n] = ite(a[m] = b[m], t[t[n]], t[e[n]])':
                                           gran('a[m] = b[m]'); gran('s[n] = t[t[n]]'); gran('s[n] = t[e[n]]');
17   case e ≡ 's[n] = t1[n] and t2[n]':    gran('s[n] = t1[n]'); gran('s[n] = t2[n]');
18   ...
19   case e ≡ 'x[n][j,i] = y[m][l,k]':     slice(x[n], j,i); slice(y[m], l,k); join(x[n]⟨j,i⟩, y[m]⟨l,k⟩);
20 }
```

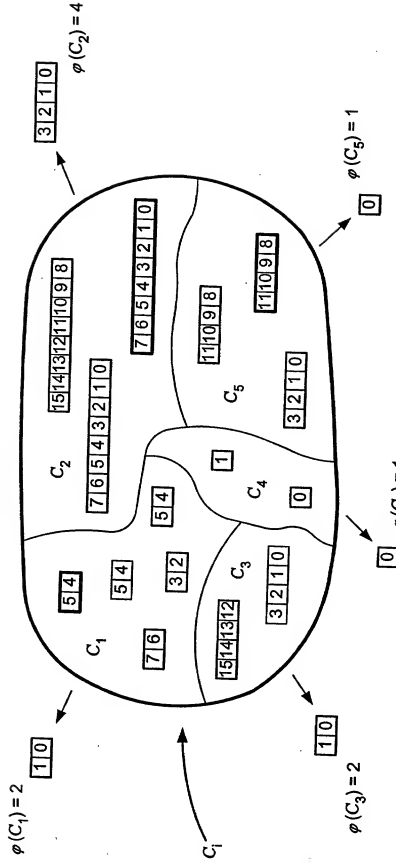


Fig. 7a

Process 2 Reduced Model Generation

```
1  for each bitvector variable x_[n] {
2    m := 0 ;
3    for each chunk x_[n]⟨j, i⟩ of the computed granularity of x_[n] {
4      C := find ( x_[n]⟨ j, i ⟩ ) ;   // equivalence class containing x_[n]⟨j,i⟩
5      m := m + φ(C) ;
6    }
7    if (m ≥ n) then m := n ;
8    replace all occurrences of x_[n] in the bitvector equations by x'_[m] ;
9    and adjust all extraction expressions affected by x_[n] ;
10 }
```

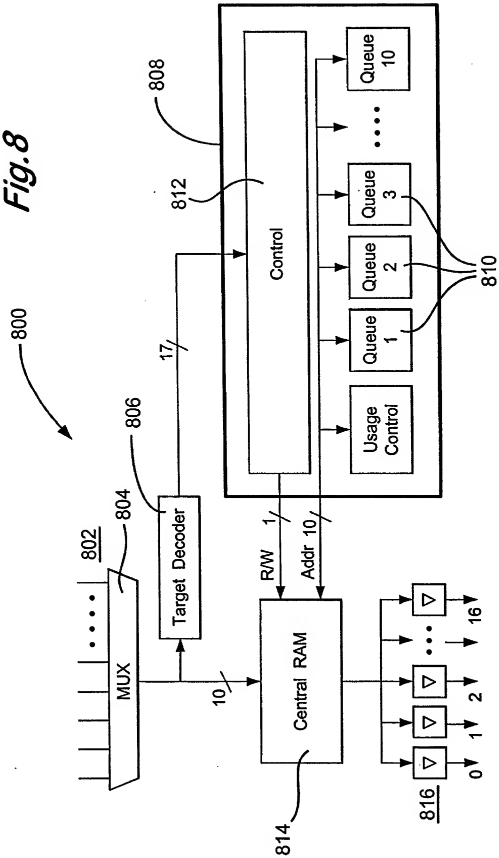

800
808
812
Control
Queue 10
Queue 3
Queue 2
Queue 1
Usage Control
810
17
806
804
802
MUX
Target Decoder
R/W
1
Addr
10
10
Central RAM
814
16
2
1
0
816

Fig.8

Fig.9

	Property	Original design	Reduced model
Computation times of the prototype for analysis and reduced model generation	nop read write		2.96 secs 6.53 secs 3.24 secs
FIFO sizes on RTL	nop read write	160 cells x 10 bit 160 cells x 10 bit 160 cells x 10 bit	160 cells x 2 bit 160 cells x 3 bit 160 cells x 3 bit
Overall number of bits in all relevant signals (cones of influence of the property)	nop read write	20925 31452 14622	5034 (24.0 %) 10592 (33.6 %) 5163 (35.3%)
Overall number of gates in synthesized netlist	nop read write	23801 23801 23801	5661 (27.9 %) 7929 (33.3 %) 7929 (33.3 %)
Number of state bits	nop read write	1658 1658 1658	362 (21.8 %) 524 (31.6 %) 524 (31.6 %)
Property checker runtimes	nop read write_fail write_hold	23:33 min 42:23 min 2:08 min 27:08 min	37.96 secs (2.7 %) 3:27 min (8.1 %) 25.66 secs (19 .5 %) 1:08 min (4.2 %)

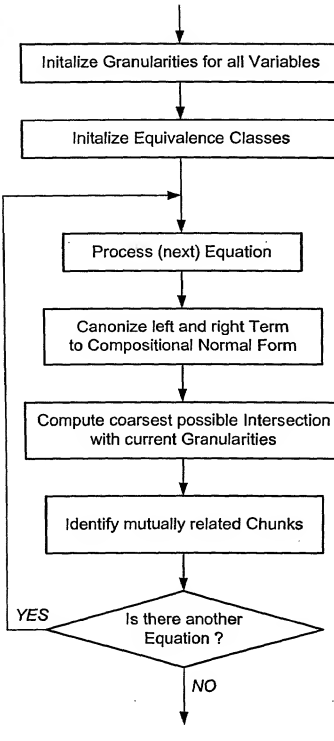


Fig. 10

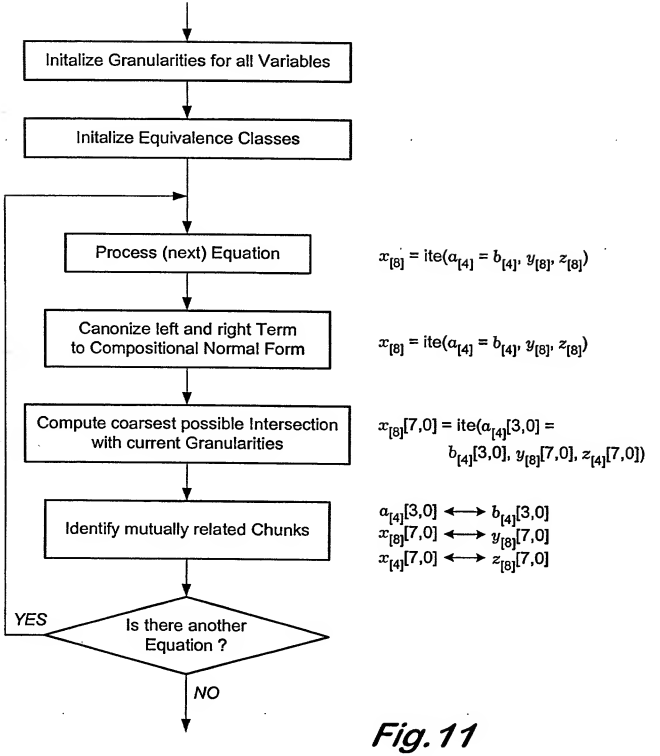


Fig. 11

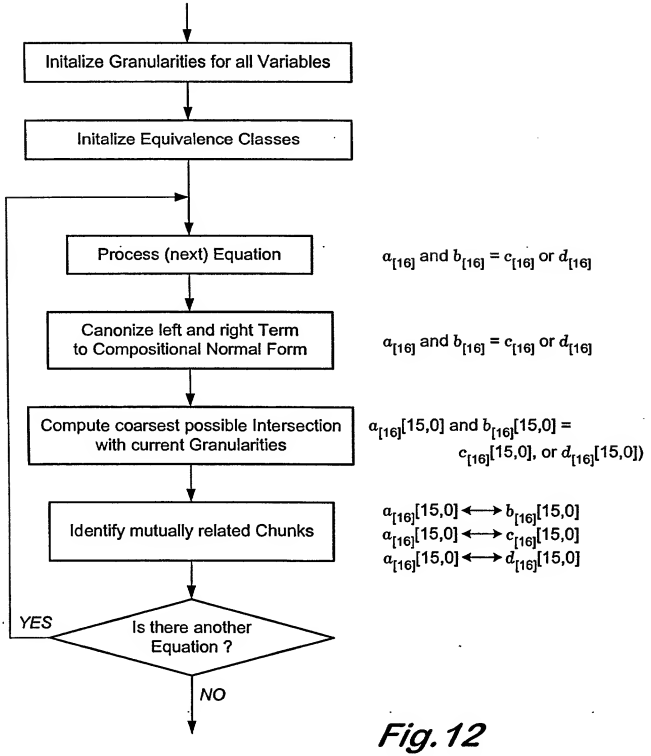


Fig. 12

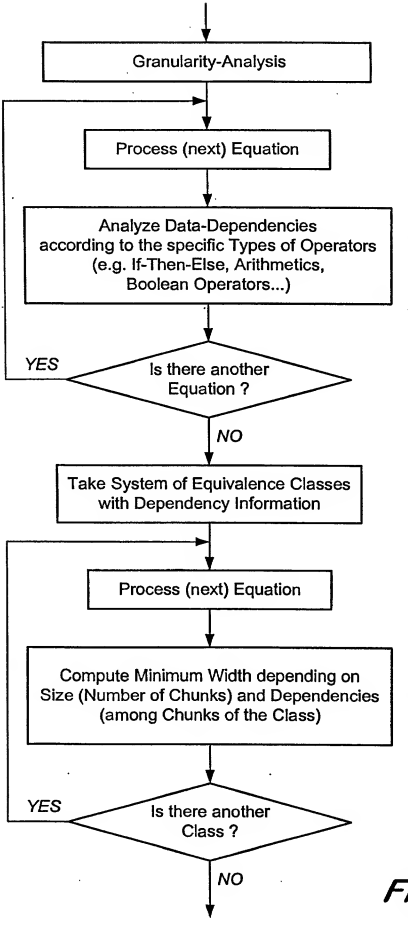


Fig. 13

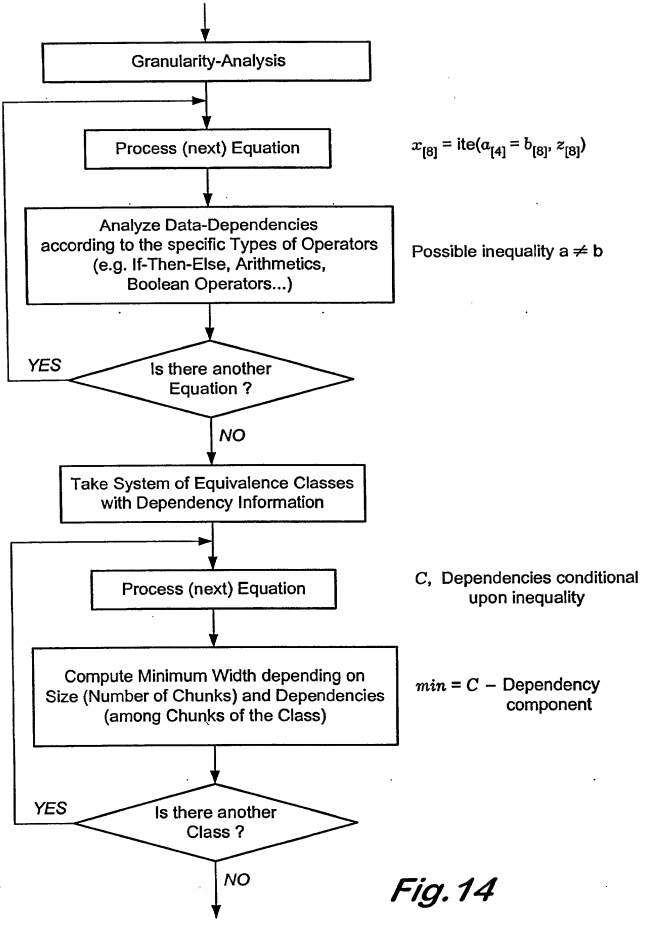


Fig. 14

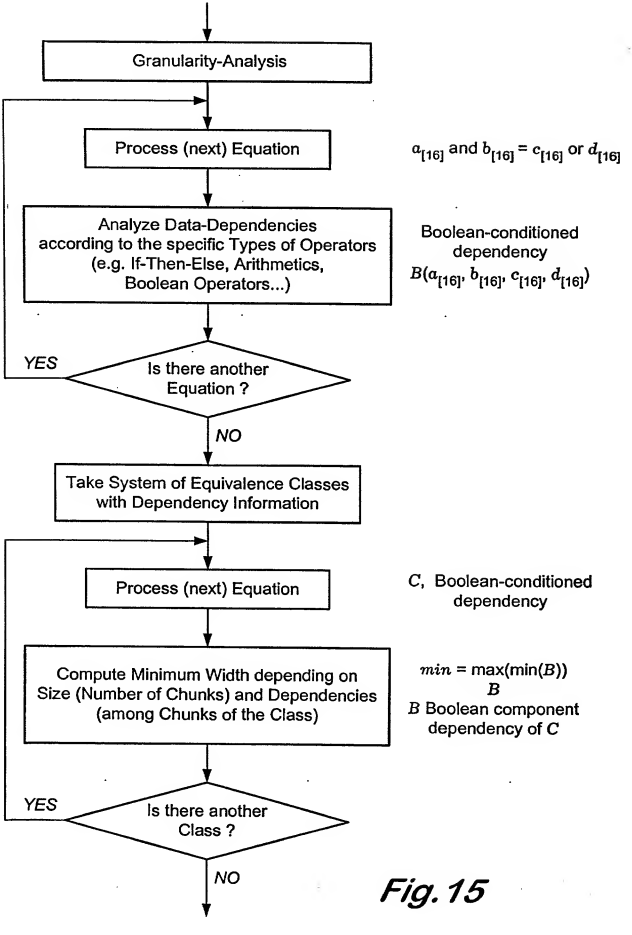


Fig. 15

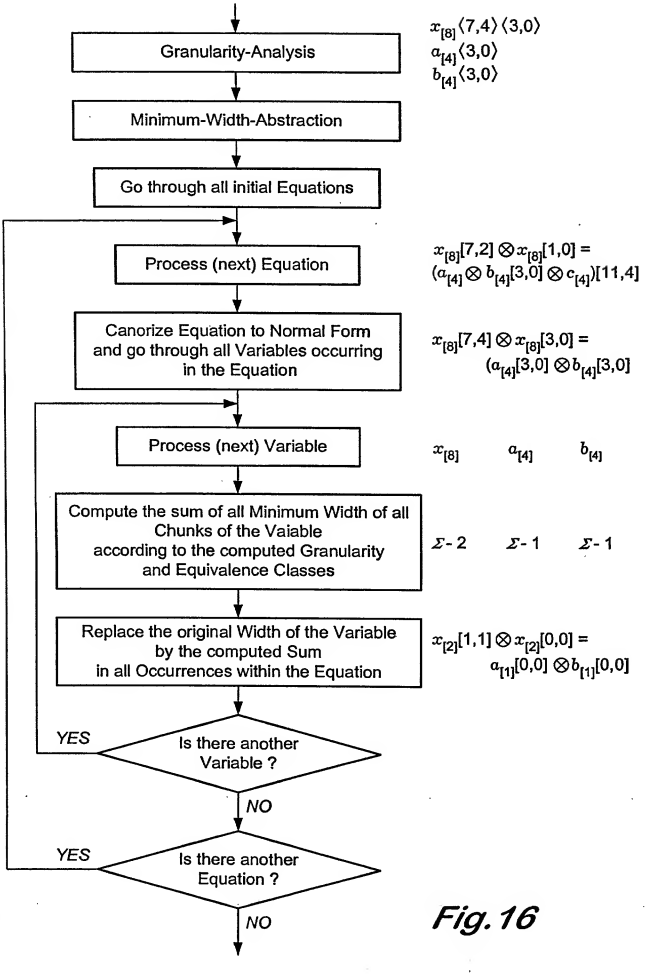


Fig. 16

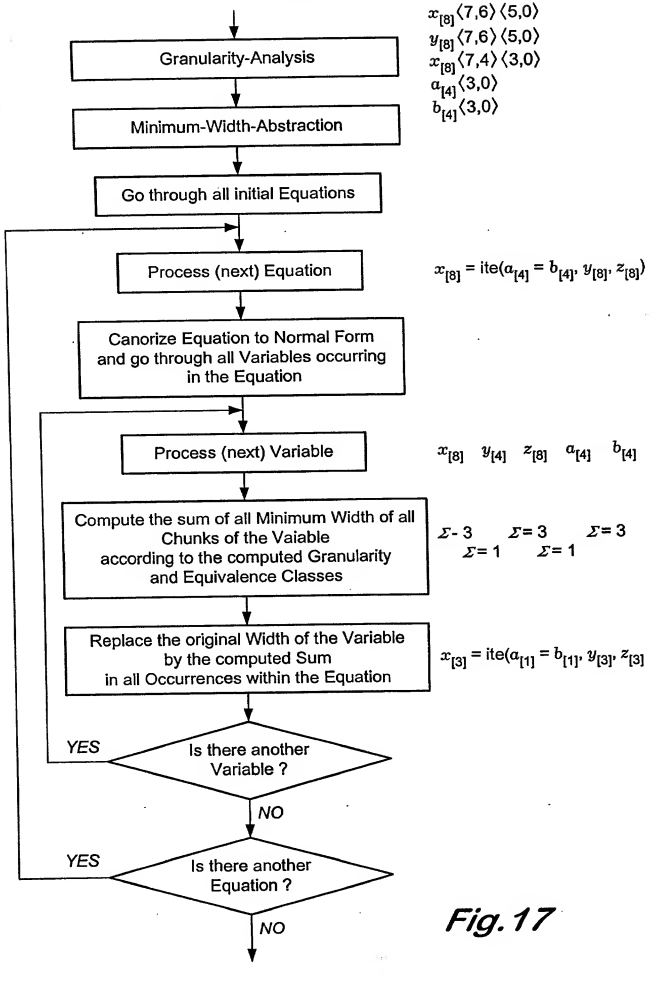


Fig. 17

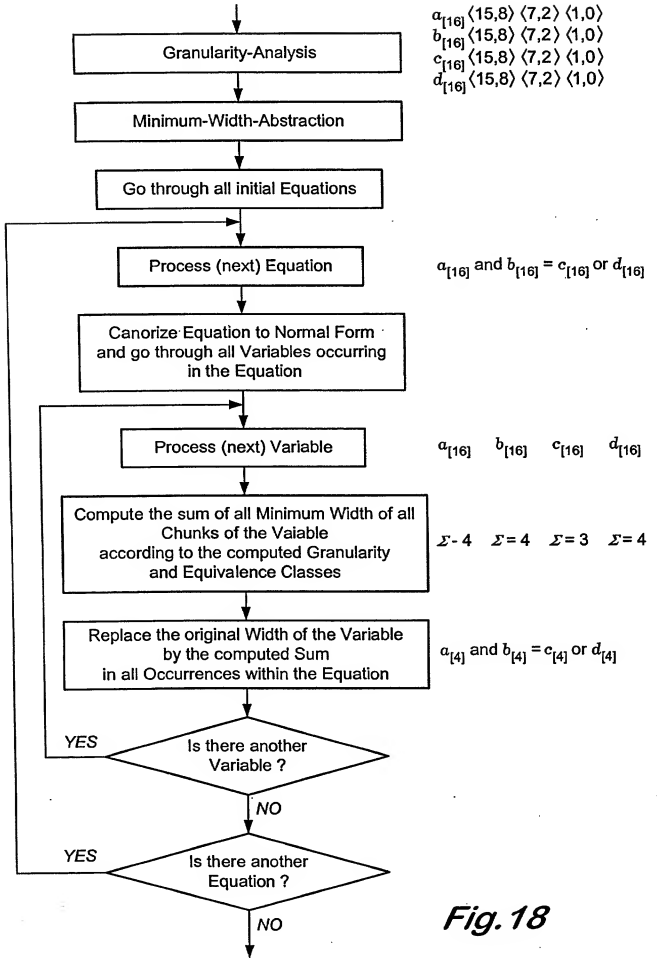


Fig. 18